

REMARKS

[0010] Applicant respectfully requests reconsideration and allowance of all of the claims of the application. The status of the claims is as follows:

- Claims 1-5 and 11-13 are currently pending
- No claims are canceled herein
- Claims 1 and 11 are amended herein

[0011] Support for the amendments to claims 1 and 11 can be found in the specification at least at pages 19-21.

Claims 1-5 and 11-13 Recite Statutory Subject Matter Under § 101

[0012] Claims 1-5 and 11-13 stand rejected under 35 U.S.C. § 101 as allegedly being directed to non-statutory subject matter. Applicant respectfully traverses this rejection.

[0013] Nevertheless, for the sole purpose of expediting prosecution and without commenting on the propriety of the Office's rejections, Applicant herein amends claims 1 and 11 as shown above. Claims 2-5 and 12-13 depend from amended claim 1 and 11, respectively, and incorporate by their dependency the amended language. Applicant respectfully submits that these amendments render the § 101 rejection moot.

Cited Documents

[0014] The following documents have been applied to reject one or more claims of the Application:

- Berkland: Berkland et al., U.S. Patent Application Publication No. 2004/0111525
- Sato: Sato et al., U.S. Patent Application Publication No. 2004/0120344

Claims 1-5 and 11-13 Are Non-Obvious Over Berkland in view of Sato

[0015] Claims 1-5 and 11-13 stand rejected under 35 U.S.C. § 103(a) as allegedly being obvious over Berkland in view of Sato. Applicant respectfully traverses the rejection.

[0016] Nevertheless, without conceding the propriety of the rejection and in the interests of expediting allowance of the application, Applicant herein amends claim 1 to incorporate features recited in the specification. Applicant submits that Berkland does not teach or suggest the following features of this claim, as amended (with emphasis added):

A method for discovering services available in a computing environment, comprising:
in an application program executed on a computer:
defining at least one discovery scope **identifying a searchable query domain**;

defining a discovery filter **comprising a simple filter and a rich filter**,
wherein the simple filter provides for expression of the searchable query domain by service type, service interface and/or service characteristics and the rich filter provides for expression of the searchable query domain using query semantics; and
initiating a search request to a first application programming interface;
in the first application programming interface:
parsing the search request;

retrieving service information corresponding to the requested discovery scope and discovery filter; and
returning the service information to the application program.

[0017] Claim 1 recites in part, “defining at least one discovery scope identifying a searchable query domain.” The Office cites Berkland as teaching “defining a discovery scope.” (Office Action, page 3.) The Office cites Figs. 5-6 in Berkland as allegedly teaching this element. Berkland, however, is directed to “a mechanism for the dynamic discovery and selection of web service implementation at runtime without explicit client control: (Berkland, Abstract.) For ease of discussion, Figs. 5 and 6 from Berkland are shown below:

500

```

510 { <wsdl:config discovery = "classNameDD" selection = "classNameSD"...>
    ...
    <services>
520 {   <service name = "name1"
        discovery = "classNameD1" selection = "classNameS1"...>
        ...
        </service>
    ...
530 {   <service name = "nameN"
        discovery = "classNameDN" selection = "classNameSN"...>
        ...
        </service>
    </services>
    ...
    </wsdl:config>
  }

```

FIG. 5

```

public Object[]
discovery(QName portTypeName,
String wsdlReference);

public Object[]
discovery(QName portTypeName,
Definition wsdlDefinition);

```

Goal: Discover services implementing a WSDL portType.

In both methods, portTypeName is a QName (qualified name with a namespace URI and a local name) for a WSDL portType. In the first method, wsdlReference is a URI pointing to a WSDL document. This is the reference to the original WSDL document describing the service (it at least includes the <portType> element) used by the WSIF client. In the second method, wsdlDefinition is a WSDL4J representation of the original WSDL document describing the service.

Both methods return an array of Object; each Object must be either a String representing a WSDL URI, or a WSDL4J Definition; either will be used in the Selection mechanism.

FIG. 6

[0018] Berkland recites in Fig. 6 that "[i]n the first method, wsdlReference is a URL pointing to a WSDL document." (Berkland, Fig. 6) Berkland further recites "[i]n the second method, wsdlDefinition is a WSDL4J representation of the original WSDL

document describing the service...Both methods return an array of Object; each Object must be either a String representing a WSDL URL, or a WDL4J Definition; either will be used in the election mechanism.” (Berkland, Fig. 6)

[0019] However, Berkland does not teach or suggest “defining at least one discovery scope **identifying a searchable query domain**,” as recited in amended claim 1. (Emphasis added.)

[0020] Claim 1 also recites in part, “defining a discovery filter comprising a simple filter and a rich filter.” The Office cites Berkland as teaching “defining a discovery filter.” (Office Action, page 3.) The Office cites Fig. 7 in Berkland as allegedly teaching this element. Again, for ease of discussion, Fig. 7 from Berkland is shown below:

```
public void selection (QName portTypeQName,
                     Object[] candidateWSDL,
                     String originalWSDLReference);
public void selection (QName portTypeQName,
                     Object[] candidateWSDL,
                     Definition originalWSDLDefinition);

public Definition getDefinition();
public Port getPort();
public Service getService();
```

Goal: Select a service implementation and a port within the service.

The selection methods initiate the actual selection mechanism. The portTypeQName in both forms is a QName for a WSDL portType. The Object array for both forms comes from the Discovery mechanism. The originalWSDLReference in the first form is a URI pointing to a WSDL document. This is the reference to the original WSDL document describing the service (it at least includes the <portType> element) used by the WSIF client. The originalWSDLDefinition is a WSDL4J representation of the original WSDL document describing the service.

The other methods return various WSDL4J classes describing

- *The entire WSDL service definition
- *The WSDL service element only
- *The WSDL port element only

FIG. 7

[0021] Berkland recites in Fig. 7 that “[t]he selection methods initiate the actual selection mechanism.” (Berkland, Fig. 7) Berkland further recites “[t]he other methods return

various WSDL4J classes describing *The entire WSDL service definition *The WSDL service element only *The WSDL port element only." (Berkland, Fig. 7)

[0022] However, Berkland does not teach or suggest "defining a discovery filter, **comprising a simple filter and a rich filter,**" as recited in amended claim 1. (Emphasis added.)

[0023] Consequently, neither Berkland nor Sato teaches or suggests all of the elements and features of this claim. Accordingly, independent claim 1 is believed to be allowable over Berkland and Sato for at least the reasons discussed.

[0024] **Dependent claims 2-5** depend from independent claim 1 and are allowable by virtue of their dependency from allowable claim 1, as well as for the additional features that each recites.

[0025] **Independent claim 11** is rejected as being obvious over Berkland in view of Sato for similar reasons as claim 1. Claim 11 is allowable at least for reasons similar to those discussed above in regards to claim 1.

[0026] **Dependent claims 12 and 13** depend from independent claim 11 and are allowable by virtue of their dependency from allowable claim 11, as well as for the additional features that each recites.

Conclusion

[0027] Applicant submits that all pending claims are in condition for allowance. Applicant respectfully requests reconsideration and prompt issuance of the application. If any issues remain that prevent issuance of this application, the Examiner is urged to contact the undersigned representative for the Applicant before issuing a subsequent Action.

Respectfully Submitted,

Lee & Hayes, PLLC
Representative for Applicant

/Dale G. Mohlenhoff 37,683/ Dated: October 8, 2009

Dale G. Mohlenhoff (dalem@leehayes.com; 509-944-4738)

Registration No. 37,683

Robert G. Hartman (rob@leehayes.com)

Registration No. 58,970